



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV INTELIGENTNÍCH SYSTÉMŮ**

DEPARTMENT OF INTELLIGENT SYSTEMS

**INTELIGENTNÍ DOPLŇOVÁNÍ SLOV V GUI PRO PŘE-  
PISOVÁNÍ MATRIČNÍCH ZÁZNAMŮ**

INTERACTIVE GUI FOR PARISH RECORDS REWRITING

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. TOMÁŠ ZATLOUKAL**

**VEDOUcí PRÁCE**

SUPERVISOR

**Ing. JAROSLAV ROZMAN, Ph.D.**

**BRNO 2019**

## Zadání diplomové práce



22006

Student: **Zatloukal Tomáš, Bc.**  
Program: Informační technologie    Obor: Inteligentní systémy  
Název: **Interaktivní GUI pro zadávání matričních záznamů**  
**Interactive GUI for Parish Records Rewriting**  
Kategorie: Umělá inteligence  
Zadání:

1. Vytvořte přehled podpůrných technik pro přepisování velkého množství dat do počítače. Seznamte se s typy a se strukturou matričních záznamů.
2. Na základě nastudovaných technik navrhnete systém, který umožní co nejrychlejší zadávání dat z matrik do počítače. Zaměřte se jak na automatické doplňování slov, tak také na rozložení prvků GUI a pohyb v něm.
3. Navržený systém implementujte jako součást GUI pro přepisování matričních záznamů.
4. Výsledný systém otestujte na skupině genealogů a zhodnoťte úsporu času zápisu pro jeden záznam. Navrhněte případná vylepšení.

### Literatura:

- Autocomplete, <https://jqueryui.com/autocomplete/>
- Lednická B., Sestavte si rodokmen, Grada, ISBN: 978-80-247-4069-0

Při obhajobě semestrální části projektu je požadováno:

- První dva body zadání

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Rozman Jaroslav, Ing., Ph.D.**

Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.

Datum zadání: 1. listopadu 2018

Datum odevzdání: 22. května 2019

Datum schválení: 1. listopadu 2018

## Abstrakt

Tato práce se zabývá zefektivněním přepisu matrik do počítače. Cílem je vytvoření mechanismů, které zlepší orientaci ve větším množství vstupů formuláře. Tyto mechanismy by také měly chytrě doplňovat vstupní pole a kontrolovat chyby.

## Abstract

This thesis deals with the efficiency of transcription of matrices into computers. The goal is to create mechanisms that will improve the orientation in a larger number of form inputs. These mechanisms should also smartly complement the input field and check for errors.

## Klíčová slova

matriky, genealogie, serielní historické prameny, webová aplikace, chytré doplňování, zadávání velkého množství dat

## Keywords

parish books, genealogy, serial sources, web application, smart suggestions, entering large amounts of data

## Citace

ZATLOUKAL, Tomáš. *Inteligentní doplňování slov v GUI pro přepisování matričních záznamů*. Brno, 2019. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Jaroslav Rozman, Ph.D.

# Intelligentní doplňování slov v GUI pro přepisování matričních záznamů

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Jaroslava Rozmana Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Tomáš Zatloukal  
21. května 2019

## Poděkování

Chtěl bych poděkovat vedoucímu práce Ing. Jaroslavu Rozmanovi, Ph.D. za odbornou pomoc a užitečné rady k řešení projektu.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>3</b>
<b>2</b>	<b>Význam matrik</b>	<b>4</b>
<b>3</b>	<b>Serielní prameny</b>	<b>5</b>
3.1	Lánové rejstříky . . . . .	5
3.2	Rektifikační akta . . . . .	5
3.3	Urbáře . . . . .	5
3.4	Opisy urbářů . . . . .	6
3.5	Sčítací operáty . . . . .	6
3.6	Matriky . . . . .	6
<b>4</b>	<b>Digitalizace matrik v česku</b>	<b>9</b>
4.1	Archiv hl.m.Prahy . . . . .	9
4.2	SOA Litoměřice . . . . .	9
4.3	ZA Opava . . . . .	9
4.4	SOA Třeboň . . . . .	9
4.5	SOA Praha . . . . .	10
4.6	SOA Plzeň . . . . .	10
4.7	MZA Brno . . . . .	10
4.8	Národní archiv . . . . .	10
4.9	Archiv města Ostravy . . . . .	10
<b>5</b>	<b>DEMoS</b>	<b>11</b>
<b>6</b>	<b>Webové technologie</b>	<b>12</b>
6.1	Nette . . . . .	12
6.2	Bootstrap . . . . .	12
6.3	JSON . . . . .	12
6.4	jQuery . . . . .	13
6.5	jQuery UI . . . . .	13
<b>7</b>	<b>Automatické doplnění s našeptávačem</b>	<b>14</b>
7.1	Autofill . . . . .	14
7.2	Google . . . . .	14
7.3	Foxentry . . . . .	16
<b>8</b>	<b>Zobrazení matrik</b>	<b>17</b>
8.1	Zoomify . . . . .	17

8.2	Star Zoom . . . . .	17
<b>9</b>	<b>Návrh</b>	<b>19</b>
9.1	Kontrola vstupů . . . . .	19
9.2	Našeptávač . . . . .	20
9.3	Přehledová mapa . . . . .	20
9.4	Pohyb po formuláři . . . . .	21
9.5	Okno s oskenovaným záznamem . . . . .	21
9.6	Fullscreen režim . . . . .	23
<b>10</b>	<b>Implementace</b>	<b>24</b>
10.1	Bootstrap 4 . . . . .	24
10.2	Kontrola vstupů . . . . .	24
10.3	Přehledová mapa . . . . .	25
10.4	Sjednocené umístění nástrojů . . . . .	26
10.5	Pohyb po formuláři . . . . .	26
10.6	Zobrazení digitalizované matriky . . . . .	27
10.7	Režim celé obrazovky . . . . .	28
10.8	Vlastní rozvržení formuláře . . . . .	29
10.9	Automatické doplnění záznamu . . . . .	30
<b>11</b>	<b>Testování</b>	<b>33</b>
11.1	Uživatelské testování . . . . .	33
11.2	Výsledky testování . . . . .	33
11.3	Zhodnocení . . . . .	34
<b>12</b>	<b>Závěr</b>	<b>36</b>
	<b>Literatura</b>	<b>37</b>
<b>A</b>	<b>Obsah přiloženého paměťového média</b>	<b>39</b>

# Kapitola 1

## Úvod

Stále větší poptávka po genealogických datech dnes naráží na překážku jejich obtížné indexace. Existuje celá řada seriálních pramenů, jako jsou matriční knihy, knihy sčítání lidu, urbáře, aj. Část těchto historických dokumentů již byla úspěšně digitalizována, stále však není možné informace jednoduše zadat do vyhledávacího pole a najít informace o svých předcích. Jedním z důvodů je skutečnost, že tuto činnost nelze provést naprosto autonomně. Záznamy jsou psány ručně v různých jazycích, písmo se mění dle toho, kdo záznam zapisoval. K efektivnímu přepisování velkého množství dat je potřeba více než formulář s připravenými vstupy pro data. Je třeba navrhnout taková vylepšení, která umožní strávit zadáním záznamu jen skutečné minimum času.

Cílem této práce je navrhnout a implementovat vylepšení vyvíjeného informačního systému pro přepisování matričních záznamů, která zefektivní samotné zadávání dat do systému. Protože je systém implementován jako webová aplikace, navržená vylepšení musejí rovněž využívat webové technologie.

V první kapitole práce popisují význam matričních záznamů a stručně jejich historii. Dále se zaměřuji na různé seriální prameny, jejich digitalizaci a u matričních knih i o indexaci. Následuje kapitola o novém informačním systému pro správu matričních záznamů DEMoS. Další kapitola je věnována návrhu jednotlivých vylepšení pro zefektivnění zadávání záznamů ve webové aplikaci DEMoS. V závěru poté uvádím pohled na další vývoj projektu vzhledem k diplomové práci.

## Kapitola 2

# Význam matrik

„Matriky jsou veřejné úřední knihy pro evidenci občanů, najdete v nich všechny důležité životopisné údaje a proto jsou matriky základním pramenem genealogického výzkumu“[21]. [1] Počátky matričních knih našem území se datuje do poloviny 16. století. První matriky měly podobu jednoduchého seznamu, který farářům dovoľoval udržovat si přehled o svém stádu. Dle toho kdo matriční knihu zapisoval byly zápisy bohaté i na další informace nebo naopak velmi strohé. V roce 1784 se staly matriky oficiálně úředním dokumentem, byla také přesně stanovena jejich struktura a obsah. Byly však stále ve správě farních úřadů, neboť ty vedly křty, svatby a pohřby. Roku 1870 vznikly vedle těchto matrik také civilní matriky vedené u okresních soudů z počátku pro bezvěrce, kteří vystoupili z církve. Později se však v těchto záznamech objevují i věřící, kteří uzavřeli civilní sňatek. V prosinci roku 1949 byl vydán zákon, který dal všechny matriky státu. Matriky tak byly dále spravovány národními výbory. Později se některé matriky dostaly do archivu a jiné zůstaly na úřadech. Dnes lze v archivu najít matriky asi do roku 1910.

[22] Matriční knihy jsou pak trojího druhu, viz následující sekce. Lze se setkat s matrikou narozených, matrikou oddaných a matrikou zesnulých. Matrika narozených obsahovala informace datu narození dítěte, jméno a příjmení křtitele, jméno dítěte, příjmení otce a jméno matky. Byla také uvedena jména kmotrů a kmoter, vyznačena zde byla osoba, která při křtu držela dítě na rukou.

V knize oddaných najdeme především datum svatby, jména a příjmení snoubenců, rodičů a svědků, jméno oddávajícího.

Nakonec v knize zemřelých lze najít datum pohřbu, jméno a příjmení zemřelého, jeho přibližný věk a jméno pohřbívajícího.



## Kapitola 3

# Serielní prameny

Mezi serielní prameny řadíme historické dokumenty jako jsou matriční knihy, pozemkové knihy, aj. V poslední době se tyto dokumenty digitalizují a tak vniká zájem o jejich indexaci do databází pro další analýzu a zpracování. Tato práce se zaměřuje především na indexaci matrik, ovšem indexace těchto dokumentů je do značné míry podobná, a tak by se získané znalosti a techniky daly využít i pro indexaci dalších pramenů. Následující sekce Obsahují základní informace o takovýchto serielních pramenech.

### 3.1 Lánové rejstříky

[13] Vůbec nejstarší dochovaný moravský katastr jsou Lánové rejstříky. Tento soupis majitelů gruntů obsahuje údaje až z roku 1657. Tvořen je souborem o 365 svazcích pro všechna moravská panství, statky včetně enkláv ležících ve Slezsku. Digitalizace těchto svazků proběhla v průběhu let 2013 a 2014. Prohlížení je možné pomocí webové aplikace případně je k dispozici ve formátu PDF. Později byl také vypracován náhled prostřednictvím webu mapy.cz. Takto je k dispozici většina digitalizovaných pramenů Moravským zemským archivem, včetně Retifikačních akta 3.2, Urbáře 3.3 a Opisy urbářů 3.4.

### 3.2 Rektifikační akta

[16] Jedná se o přípravný materiál k tereziánskému rustikálnímu katastru. Vznik toho pramenu je datován mezi roky 1749-1768 a starší berní písemnosti z období 1667-1768. Mimo jména držitelů pozemků, údaje o výměře polí, vinic a dalších kultur, jsou zde také uvedena jména řemeslníků a jejich řemesel. Jedná se o cenný historický materiál i z toho důvodu, že umožňuje najít shodu s držiteli gruntů z doby lánové vizitace z podkapitoly 3.1.

### 3.3 Urbáře

[18] Jsou to písemnosti vytvářené vrchností pro evidenci dávek, které jim byly nebo měly být vypláceny poddanským obyvatelstvem. Později jejich funkci převzaly každoroční rejstříky pozemků. Na Moravě jsou předmětem soupisu urbáře vzniklé před rokem 1775, kdy byly poddanské povinnosti vůči vrchnosti a kostelu stanoveny státem. Digitalizovány jsou průběžně od roku 2012.

### 3.4 Opisy urbářů

[15] Od bývalého školského inspektora PhDr. Jaroslava Novotného získal Moravský zemský archiv roku 1963 jeho rukopisnou sbírku opisů moravských urbářů a odhadů. Sbírká čítá na 616 sešitů a více než 13 500 stran obsahující informace o téměř všech známých urbářů a odhadů k dějinám moravských panství od roku 1750. Díky obsahu dnes již nedostupných opisů urbářů je tato sbírka velice cenná. Digitalizace této sbírky započala v roce 2015.

### 3.5 Sčítací operáty

[17] Reskriptem vydaným 13. října 1753 Marií Terezií bylo ustanoveno pravidelné sčítání lidu v Habsburské monarchii. O rok později proběhlo první sčítání. Důležitým milníkem je však rok 1869, kdy v monarchii došlo k prvnímu modernímu sčítání lidu, a také bylo stanoveno opakování sčítání pravidelně každých deset let. Digitalizace sčítacích operátů započalo v roce 2014 ve spolupráci s organizací FamilySearch viz. sekce 3.6.1.

### 3.6 Matriky

Přepisování matrik je činnost, při které se originální ale častěji digitalizované matriky přepisují do počítače. Tuto činnost musí provádět člověk, i když už tu byly pokusy jak jej alespoň částečně vynechat. K dosažení kvalitních výsledků je vždy potřeba, aby přepsání provedl odborník, to je ale časově náročné. Často se proto setkáváme s metodami, kdy přepis obstarávají dobrovolníci a odborníci kontrolují jejich výsledky.

#### 3.6.1 FamilySearch

[6] Historicky známý jako Genealogical Society of Utah, si roku 1894 stanovil za cíl zachovat historické rodinné záznamy lidstva. Nyní pod záštitou církve Ježíše Krista, Svatých posledních dnů. Obecně je církev spíše známá pod jménem Mormoni. FamilySearch dává k dispozici jak matriční záznamy tak i techniku, jak data přepsat do databáze. Vše je zdarma přístupné na oficiální webové adrese projektu.

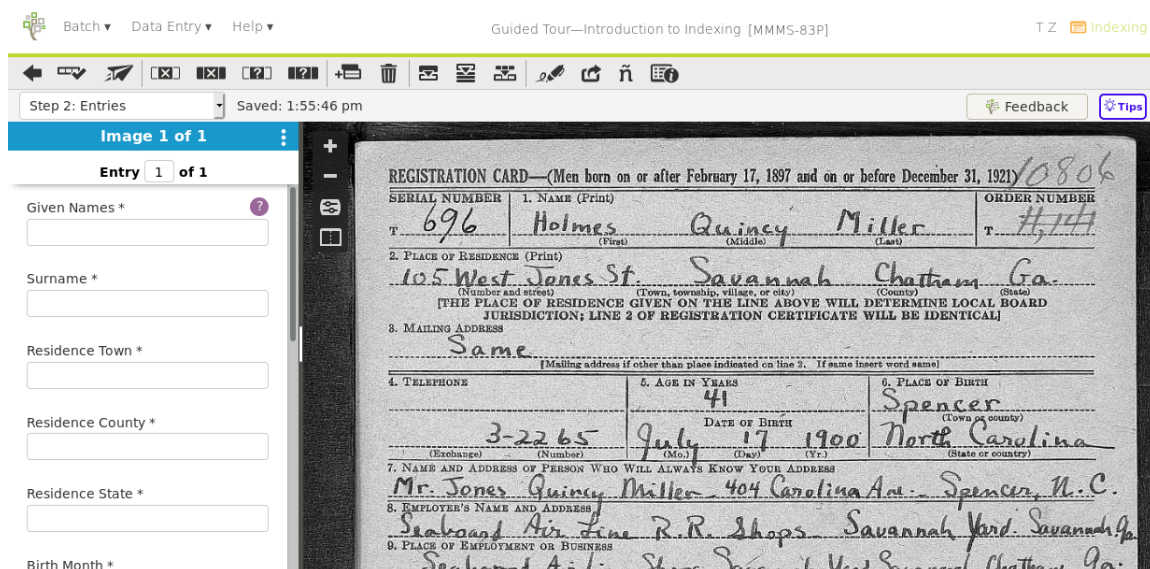
[6] Jakmile je k dispozici digitální podoba matričního záznamu mohou dobrovolníci z celého světa data přepisovat do připravených formulářů. Přepsaná data poté kontrolují pracovníci FamilySearch. Po schválení jsou data publikována, lze v nich vyhledávat a sestavovat rodokmeny.

Za dobu své existence prošlo prostředí pro indexování řadou změn a vylepšení celého uživatelského rozhraní. Největší část obrazovky je věnována digitalizovanému záznamu 3.1. Formulář pro přepis dat se nachází na levé straně. V horní liště se pak nachází užitečné funkce pro vytvoření či naopak smazání vstupního pole, označení pole jako prázdné nebo nečitelné. Užitečná je i funkce, která zobrazí příklady ručně psaných znaků abecedy 3.2.

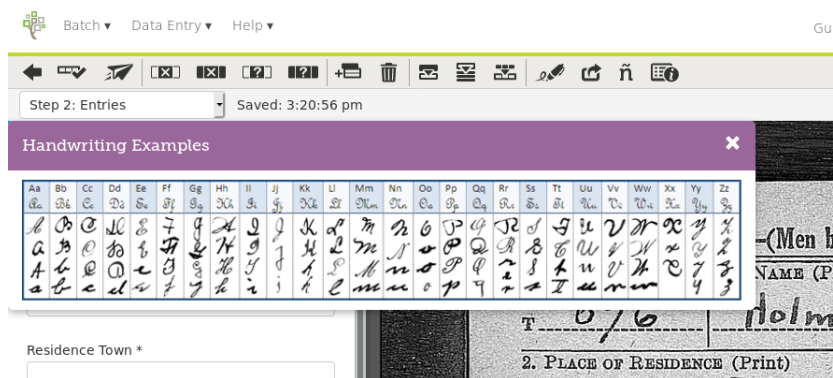
#### 3.6.2 Crowdsourcing

V roce 2014 byla vydána práce [7], která spojovala počítačové vidění a crowdsourcing za účelem přepisování matričních záznamů. Řešení se sice soustředilo pouze na matriky oddaných, ale řešení bylo aplikovatelné i na jiné typy dokumentů.

V první fázi bylo použité počítačové vidění pro nalezení jednotlivých slov v naskenovaných matričních záznamech. Tato slova jsou jako obrázek náhodně předkládána uživatelům,



Obrázek 3.1: Uživatelské rozhraní pro indexování na portálu FamilySearch.org

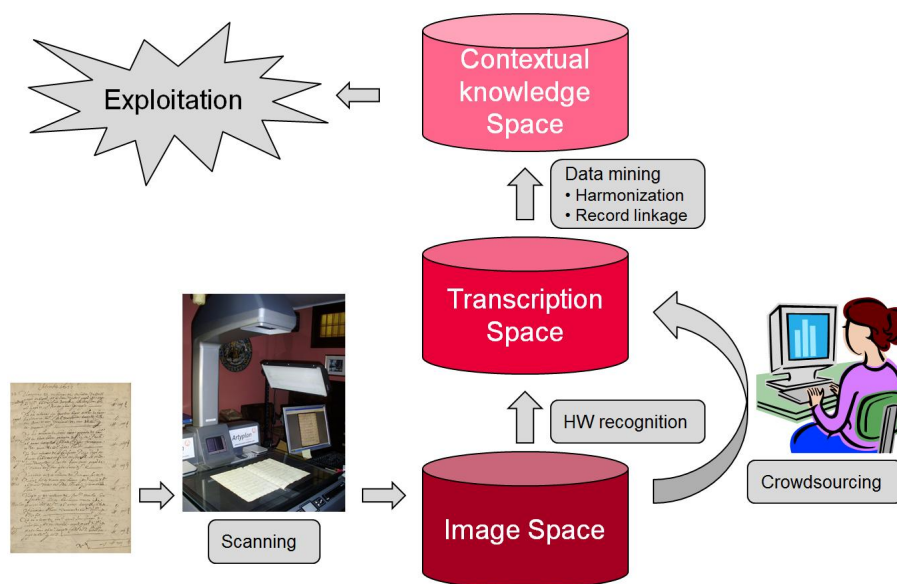


Obrázek 3.2: Funkce pro zobrazení příkladu psacího písma.

kterí je přepisují do textové podoby. Využívá se zde rozložení složitého úkolu na více jednodušších úkolů, ke kterému není potřeba žádné větší zaškolení a mohou tak na této platformě přispívat dobrovolníci z celého světa.

Druhý typ uživatele provádí analýzu dokumentů. Má dostupný přehled chybovostí a skóre jednotlivých slov. Z této pozice pak dává těmto slovům význam tím, že je spojuje s osobami. Mezi tato slova tedy patří například jména rodičů, jejich povolání nebo kolik zaplatili farnosti.

Touto metodou bylo dosaženo kvalitních výsledků. Za pouhé dva roky se takto povedlo přepsat za pomoci asi 150 uživatelů 287 svazků, což činí asi 600 000 záznamů.



Obrázek 3.3: Crowdsourcingová architektura aplikace [7].

## Kapitola 4

# Digitalizace matrik v česku

Digitalizace matrik v České republice začala v roce 2007. Většinu projektů digitalizace iniciovala americká genealogická společnost Family Search [3.6.1](#). Je důležité zdůraznit, že digitalizací se zde rozumí poskytnutí náhledů matrik v digitální podobě. Jedná se tedy o naskenované snímky, ve kterých nelze vyhledávat konkrétní informace.

### 4.1 Archiv hl.m.Prahy

[10] V archivu hlavního města Prahy bylo doposud digitalizováno asi 2 812 matrik od roku 1625 do 1925. Mimo matriky portál obsahuje také různé kroniky a soupis obyvatelstva. O zobrazení se zde stará knihovna Zoomify, která umožňuje obrázky v pohodlí prohlížeče přiblížit.

### 4.2 SOA Litoměřice

[24] Kromě téměř 13 tisíc digitalizovaných matrik, obsahuje státní oblastní archiv v Litoměřicích také digitalizované sčítací operáty, kroniky, úřední knihy, rukopisy a další dokumenty. Lze v tomto archivu dohledat digitalizované matriky z roku 1558 až do roku 1950. Pro samotné zobrazení zde byla opět použita knihovna Zoomify.

### 4.3 ZA Opava

[27] Zemský archiv v Opavě digitalizoval 19 189 matrik a 242 952 sčítacích operátů. Samotný portál digitálního archivu je funkčně i vizuálně téměř totožný s digitálním archivem SOA Litoměřice [4.2](#). Díky velkému množství různých filtrů a internímu vyhledávači lze matriky a další dokumenty snadno dohledat.

### 4.4 SOA Třeboň

[9] Sbírka státního oblastního archivu v Třeboni eviduje 6 875 svazků matrik v časovém rozsahu let 1587-1949. Kromě 3 nejstarších matrik jižních Čech ještě z konce 16. století jsou v tomto archivu také sčítací archy, urbáře a další dokumenty. Náhled dokumentů je zde zprostředkován javascriptovou knihovnou Zoomify.

## 4.5 SOA Praha

[9] Státní oblastní archiv v Praze se v rámci digitalizace soustřeďuje především na matriky. V archivu se jich nachází přes 10 tisíc. V roce 2012 bylo dokončeno zveřejnění 3 500 digitalizovaných matrik z mikrofilmů. Protože tato technika produkuje méně čitelné výsledky, rozhodli se digitalizovat některé matriky znovu. Jedná se o starší matriky ze 17. a 18. století. Od roku 2014 jsou náhledy přístupné na portálu eBadatelna. Součástí jsou základní informace o matrice a vyhledání matriky podle lokality či farnosti.

Další listiny tohoto archivu jsou zveřejňovány prostřednictvím mezinárodních projektu jako je například Monasterium.

## 4.6 SOA Plzeň

[25] Digitální archiv státního oblastního archivu v Plzni zpřístupňuje matriky, kroniky, fotografie, listiny, sčítací operáty a periodika, připravuje i zpřístupnění urbářů, pozemkových knih a dalších materiálů na portálu portafontium.eu. Digitalizované listiny pak postupně zpřístupňuje na stránkách projektu Monasterium.

Na webových stránkách portafontium.eu jsou sbírky matrik od roku 1581 až do roku 1945. Pro práci s obrázkovými náhledy matrik zde byla použita javaskriptová knihovna IIPIImage.

## 4.7 MZA Brno

[14] Moravský zemský archiv v Brně má své digitalizované matriky umístěny na portálu actapublica.eu. Tento projekt vznikl ve spolupráci s jedním z Rakouských archivů a je spolufinancován z Evropské unie. Rozsahem pokrývají matriky celý jihomoravský kraj. V současné době mají všechny matriky digitální podobu. Nejstarší matriky jsou z období 16. století, každoroční přírůstky jsou rovněž digitalizovány a průběžně zveřejňovány.

Portál bohužel nepoužívá pro zobrazení náhledů HTML5 technologie ale starý, dnes už zapovězený Adobe Flash.

## 4.8 Národní archiv

[20] Národní archiv v současnosti pečuje o více než 1700 archivních fondů a sbírek. Archiv se zaměřuje především na židovské matriky, kde nejstarší je z roku 1780. Digitalizované dokumenty se nachází na portálu badatelna.eu. Mimo samotný národní archiv zde lze vyhledávat v archivu Ministerstva zahraničních věcí, archivu Univerzity Karlovy, knihovny Národního muzea nebo Památníku národního písemnictví. Samotné zobrazení náhledů je zprostředkováno knihovnou Zoomify.

## 4.9 Archiv města Ostravy

[5] Digitální badatelna Archivu města Ostravy zpřístupňuje digitalizované mapy, fotografie, operáty sčítání lidu a inventáře. Pro použití digitálního archivu je třeba se nejprve registrovat. Po té lze prohlížet matriky od roku 1980.

## Kapitola 5

# DEMoS

[26] V březnu roku 2018 byl Technologickou agenturou České republiky vypsán projekt pod názvem „Možnosti tvorby komunitní genealogické databáze se sémantickou informací a zahrnutou neurčitostí“. Projekt si klade za cíl vytvořit systém pro zadávání informací ze seriálních pramenů (matriky, pozemkové knihy ad.). Systém musí být dostatečně komfortní, aby bylo zadávání dat co možná nejpohodlnější. Součástí systému je také možnost generovat rodokmeny a další data pro historicko-statistický a demografický výzkum. Na tomto projektu spolupracuje Vysoké učení technické v Brně spolu s Masarykovou Univerzitou.

Začala tak vznikat webová aplikace DEMoS (Database of early modern sources). Na první pohled vzdáleně připomíná bakalářskou práci Davida Czernína „Uživatelské rozhraní pro přepis matrik“ z roku 2018 [4], ze které také vychází. Byla však dle potřeb postupně upravována jak po stránce backendu tak frontendu. Psána je jako její předloha za použití PHP frameworku Nette pro backend.

Na aplikaci se stále intenzivně pracuje, již dnes však obsahuje základní důležité funkce. Aktuálně aplikace obsahuje rozcestník matrik dle geografické polohy, vyhledávač matrik a formulář pro přidání matričního záznamu, viz obrázek 5.1.

The screenshot shows the DEMoS web application interface. At the top, there is a header with the DEMoS logo, a search bar labeled "Vyhledávej podle obce či signatury", and navigation links for "Sutterlin", "Přihlášení", and "Nápověda". Below the header, the main section is titled "Matriky". The form for adding a church record is displayed, consisting of several panels with expandable sections (indicated by arrows). The panels include: "Pozice" (Location) with fields for Archiv, Fond, Signatura, Pořadí scenu, Rozložení na scenu, Přes celou, and Pořadí záznamu; "Datum a adresa" (Date and address) with fields for Datum narození, Datum křtu, Obec, Ulice, and Celo popisné; "Křestitel" (Godparent) with fields for Jméno, Příjmení, and Titul; "Porodní bába" (Midwife) with fields for Jméno, Příjmení, Obec, Ulice, and Celo popisné; "Dítě" (Child) with fields for Jméno, Příjmení, Vicerata, Pohlaví, Muž, Loba, Manželská, Vyznání, katolík, Datum sňatku rodičů, Mrtvé rozené, and Nalezenec; "Otec" (Father) with fields for Mrtv and Jméno; "Otcův otec" (Grandfather) with fields for Mrtv and Jméno; "Otcova matka" (Grandmother) with fields for Mrtvá and Jméno; "Otcův děda" (Great-grandfather) with fields for Jméno; and "Otcova bába" (Great-grandmother) with fields for Jméno. Each panel has a small question mark icon in the top right corner.

Obrázek 5.1: Snímek aplikace demos - formulář pro přidání matričního záznamu.



## Kapitola 6

# Webové technologie

V této kapitole popisují webové technologie, které byly v této práci použity. Protože se tato práce věnuje webové aplikaci, patří mezi základní technologie jazyk HTML, pro modifikaci vzhledu CSS a pro dynamické chování aplikace Javascript. Hlavní technologii je zde PHP framework Nette. Dále pak front-end knihovna Bootstrap. Javascriptová knihovna JQuery a její nádstavba JQuery UI.

### 6.1 Nette

[19] Nette je český open source PHP framework skládající se z řady samostatně použitelných balíčků. Tyto balíčky slouží například k práci s databází, zapouzdření HTTP požadavků, bezpečnost webových formulářů, šablonovací systém, odesílání e-mailů nebo správu přístupových práv.

Nette je aktivně vyvíjeno již přes deset let. Za tuto dobu si jeho knihovny získaly pověst dobrou pověst díky své stabilitě a bezpečnosti. V roce 2015 se mezi PHP frameworky umístil na třetím místě hned po Symfony2 a Laravelu, který anketu vyhrál.

### 6.2 Bootstrap

[12] Dnes nejpopulárnější open source front-end framework původně navržen a vytvořen společností Twitter v roce 2010. Použití tohoto frameworku velmi ulehčuje a zrychluje vývoj webových aplikací. Dodržením správných postupů lze získat responzivní, mobile-first webovou aplikaci. Pomocí HTML, CSS a Javascriptu se Bootstrap postará o vzhled a chování tlačítek, formulářů, menu, pokročilé možnosti typografie a dalších interaktivních komponent jako jsou například modální okna.

### 6.3 JSON

[3] JSON (JavaScript Object Notation) je jednoduchý datový formát pro výměnu dat, jehož původní specifikaci popsal Douglas Crockford. Je založen na podmnožině JavaSkriptu a je plně jazykově nezávislý. Oproti jiným formátům jako je například XML je jeho generování jednodušší a také je pro člověka lépe čitelný. Má vysoké uplatnění nejen ve webových aplikacích, kde je velmi často používán při komunikaci mezi prohlížečem klienta a serverem.



Data jsou popisována pomocí dvou základních struktur. Tou první je kolekce dvojic název a hodnota, s nimiž je možné se u jiných jazyků setkat pod názvy struktura či objekt. Druhou tvoří seřazený seznam hodnot, který je v různých jazycích realizován jako pole.

Dalším důvodem jeho rozšířenosti ve webových aplikacích je jeho podpora v JavaSkriptu. Funkce `JSON.stringify()` převede JavaSkriptový objekt na řetězec JSON. Naopak funkce `JSON.parse()` dokáže převést textový řetězec ve formátu JSON na JavaSkriptový objekt.

## 6.4 jQuery

[11] jQuery je nejznámější a nejpoužívanější JavaSkriptová knihovna. Zaměřuje se na manipulaci s DOM (Document Object Model), práci s událostmi, tvorbu animací a jednoduší práci s asynchroní komunikací se serverem (Ajax). Přitom zajišťuje korektní chování svých funkcí napříč prohlížeči.

## 6.5 jQuery UI

jQuery UI<sup>1</sup> je nástavba nad jQuery, která se zaměřuje na uživatelskou interakci s webovou aplikací. Řada komponent, které jQuery UI za vývojáře řešila, je dnes možné implementovat za pomoci HTML5 a CSS3. Například výběr kalendářního data, výběr čísla, implementaci drag and drop či různé efekty a animace. Další komponenty mají přímou konkurenci v Bootstrapu, popsaném v předchozí sekci 6.2, kde jsou implementovány například tlačítka, dialogová okna, menu nebo záložky. Ne však zdaleka všechny mají svojí nativní alternativu. Jednou z takových komponent je Autocomplete Widget.

### 6.5.1 Autocomplete Widget

Je jednou z hojně využívaných komponent jQuery UI. Umožňuje uživateli rychle najít a vybrat prvek z předpřipraveného seznamu hodnot, který lze během psaní do vstupního pole dále aktualizovat.

Tento widget disponuje dobře zdokumentovaným API<sup>2</sup>, které dovoluje práci našeptávače do velké míry upravovat a přizpůsobit potřebám vyvíjené webové aplikace. Lze přidat CSS třídy a upravovat tak vzhled, přepsat různé vnitřní funkce například pro vykreslení jednotlivých položek našeptávače či změnit akci po vybrání položky.

Zdroj dat pro položky našeptávače lze definovat jako pole hodnot, textový řetězec specifikující url adresu zdroje nebo vlastní funkci. Poslední možnost nabízí nejvíce flexibility, neboť lze takto našeptávač připojit k nejrůznějším zdrojům dat, a také přímo ovlivnit jaká data se mohou na server zasílat.

---

<sup>1</sup><https://jqueryui.com>

<sup>2</sup><https://api.jqueryui.com/autocomplete/>

## Kapitola 7

# Automatické doplnění s našeptávačem

V této kapitole jsem se zaměřil na zmapování dnes dostupných metod o automatickém doplnění formulářů za pomoci inteligentních našeptávačů. První z nich popisuje možnosti v rámci čistého HTML, druhé popisuje řešení Googlu pro doplnění adres a poslední je produkt společnosti Foxentry pro automatické vyplnění formuláře.

### 7.1 Autofill

V roce 2017 vydalo World Wide Web Consortium dodatečnou specifikaci pro atribut vstupních polí formuláře [2]. Doposud bylo možné nastavit atribut `autofill` na hodnotu `on` nebo `off`. Dle této hodnoty prohlížeč zobrazoval nebo nezobrazoval vlastní našeptávač dle již zadaných hodnot uživatele v jiných formulářích.

Dle této specifikace se tento atribut dá dále nastavit na další hodnoty. Podle nastavené hodnoty pak prohlížeč dokáže přesněji rozhodnout jaká data do našeptávače vložit. Těmito atributy se tak dá například nastavit specifický výběr pro celé jméno, titul nebo oslovení, položky adresy, položky platební karty, datum narození a další. Příklad použití tohoto atributu lze vidět na obrázku 7.1.

Toto jistě ocení uživatelé při vyplňování formuláře na webech elektronických obchodů. Dle Googlu, který tuto funkcionalitu začal ve svém prohlížeči Chrome podporovat ve verzi 43, jsou uživatelé schopni vyplnit formulář až o 30% rychleji.

V aplikaci DEMoS však tento atribut není vhodné použít neboť uživatel nebude opakovaně zadávat své údaje, ale opisovat údaje z matričních záznamů.

### 7.2 Google

Společnost Google nabízí vlastní JavaScriptové API<sup>1</sup> pro automatické doplnění adresy z našeptávače. Díky svému produktu Google mapy má společnost dostatek dat k přesnému vyplnění všech polí různých adres. Jakmile uživatel začne psát část adresy do pole využívající tohoto API, zobrazí se mu našeptávač 7.2. Ten obsahuje krom hodnoty pro aktuálně vyplňované pole i další související položky. Na obrázku 7.2 lze vidět, že uživatel zadává ulici. Kromě ulice však našeptávač zobrazuje i další možnosti pro stát město a zemi. Po zvolení vhodné položky jsou automaticky vyplněna i další pole, viz obrázek 7.3.

<sup>1</sup><https://developers.google.com/maps/documentation/javascript/places-autocomplete>

```

<label for="frmAddressS">Address</label>
<input name="ship-address" required id="frmAddressS" placeholder="123 Any Street" autocomplete="shipping street-address">
<label for="frmCityS">City</label>
<input name="ship-city" required id="frmCityS" placeholder="New York" autocomplete="shipping locality">
<label for="frmStateS">State</label>
<input name="ship-state" required id="frmStateS" placeholder="NY" autocomplete="shipping region">
<label for="frmZipS">Zip</label>
<input name="ship-zip" required id="frmZipS" placeholder="10011" autocomplete="shipping postal-code">
<label for="frmCountryS">Country</label>
<input name="ship-country" required id="frmCountryS" placeholder="USA" autocomplete="shipping country">

```

Obrázek 7.1: Příklad použití atributu autocomplete.

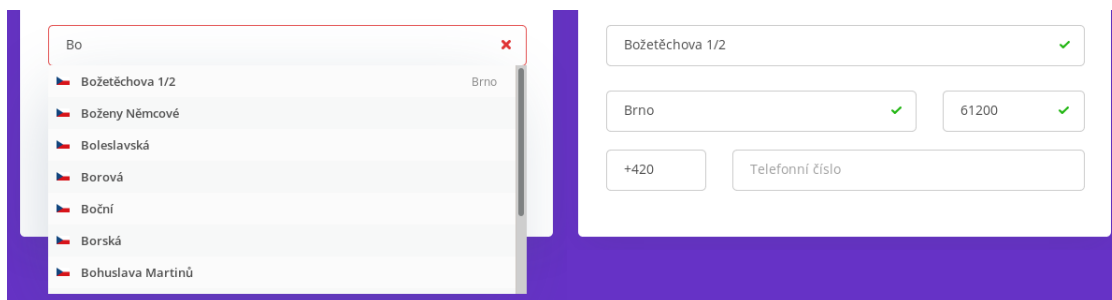
Obrázek 7.2: Ukázka našeptávače adres od Googlu. Převzato z [8].

Obrázek 7.3: Ukázka vyplněného formuláře pomocí našeptávače Googlu. Převzato z [8].

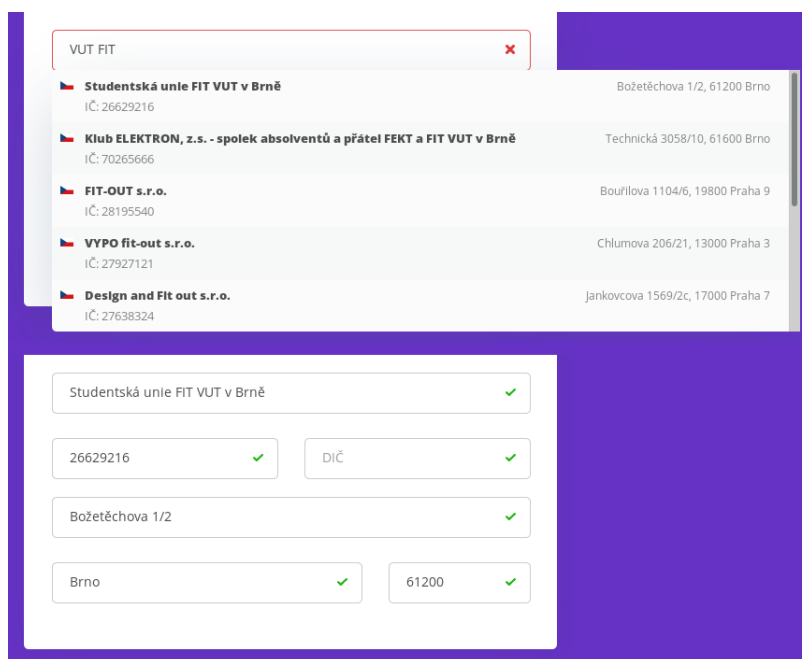
Pro účely aplikace DEMoS není však toto API vhodné, neboť uživatel bude zadávat adresy, které již nemusely existovat, a tak by se nezobrazily ani v tomto našeptávači.

## 7.3 Foxentry

Foxentry<sup>2</sup> je produkt české firmy, který si klade za cíl ukončit zbytečně zdlouhavé vyplňování formulářů na webech. Tento nástroj byl spuštěn koncem června v roce 2018. Kromě validace telefonních čísel, e-mailových adres či kontroly překlepů, nástroj disponuje inteligentním našeptávačem adres. Funguje velmi podobně jako našeptávač adres Googlu z předchozí sekce 7.2, jak je vidět na obrázku 7.4. Nástroj se však nezaměřuje čistě na adresy, ale dokáže uživateli našeptat adresu získat i při částečném vyplnění názvu firmy nebo např. identifikačního čísla osoby (IČ). Toto je lze vidět na obrázku 7.5.



Obrázek 7.4: Ukázka našeptávače adres Foxentry.



Obrázek 7.5: Ukázka našeptávače firem Foxentry.

---

<sup>2</sup><https://foxentry.cz/>

## Kapitola 8

# Zobrazení matrik

Nativní podpora pro zobrazení obrázků na webech s funkcí obraz zvětšit či zmenšit není. K docílení této funkcionality se nejčastěji používá javascript. Jak bylo zmíněno v kapitole 4 většina webových stránek s digitalizovanými matrikami užívá k jejich zobrazení uživateli javascriptovou knihovnu Zoomify. Možných řešení je však celá řada. V této kapitole se zaměřuji ne jen na ni, ale i na její možnou alternativu Star zoom.

### 8.1 Zoomify

Zoomify<sup>1</sup> je produkt společnosti Zoomify Inc. pro pokročilé zobrazení vysoce detailních fotografií na webových stránkách. K jejímu použití je třeba předpřipravit obrázky speciálním softwarem **Zoomify Free Converter** který svůj výstup nahraje do nového adresáře. Následuje klasické vložení skriptu do HTML stránky a minimální konfigurace skriptu. Ve verzi zdarma zahrnuje základní funkcionality, která zahrnuje možnost zvětšit či zmenšit obrázek nebo jej posunout.

Pro další funkcionality je třeba využít již placených verzí Zoomify Express, Zoomify Pro, nebo Zoomify Enterprise. Pak je možné získat například pokročilejší nástrojový panel, galerii obrázků, animace, anotace, filtry a další. Tyto pokročilejší nástroje jsou zobrazeny na obrázku 8.1, především lze vidět spodní nástrojovou lištu s tlačítky ke změně přiblížení fotografie a tlačítka pro navigaci v galerii, která je zobrazena v horní části obrázku. Zoomify v této verzi také zobrazuje náhled originální fotografie s obdélníkem vyznačujícím právě zobrazovanou část přiblížené fotografie. Tento prvek se nachází v pravém horním rohu elementu s fotografií, je však na obrázku 8.1 skryt pod galerií fotek. Na obrázku je překryt galerií fotek.

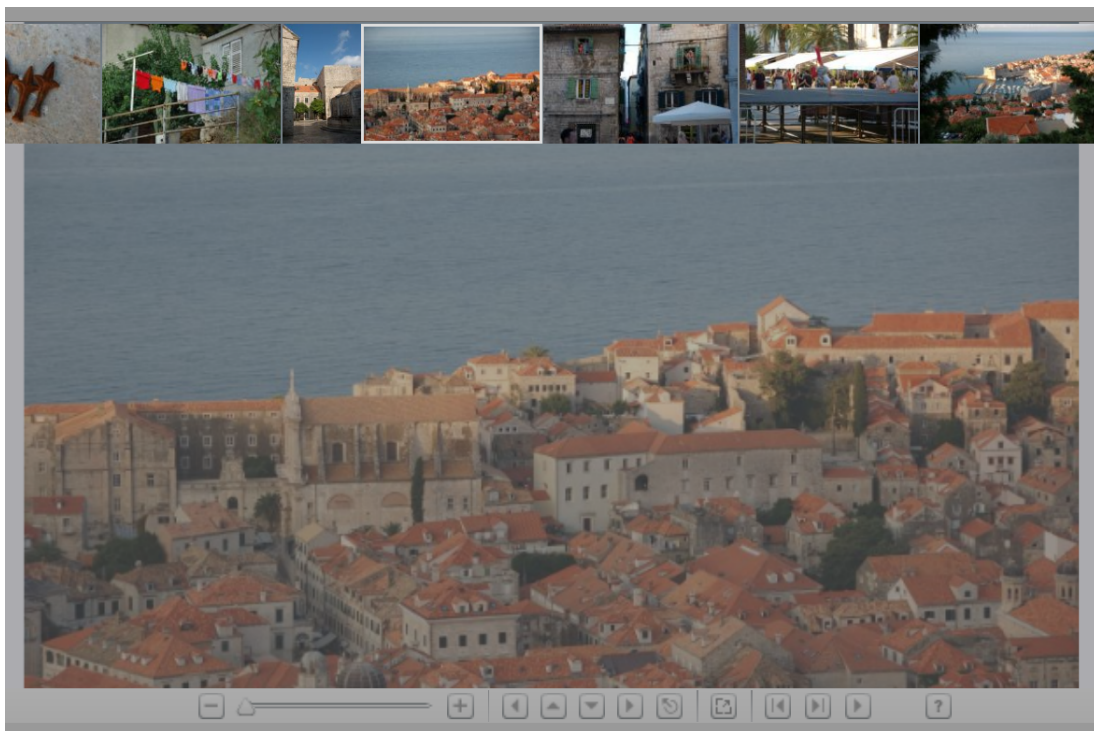
### 8.2 Star Zoom

Star Zoom<sup>2</sup> je jedním z produktů společnosti Star Plugins pro zobrazení obrázků na webových stránkách. Chlubí se především širokou podporou webových prohlížečů a systémů android či iOS, podporou detailních obrázků, jednoduchostí nasazení tohoto řešení na web a svojí rychlostí. Nedisponuje však žádnými dodatečnými funkcemi, jako je tomu v placených verzích Zoomify, v nástrojovém panelu, cílí především na jednoduchost.

---

<sup>1</sup><http://zoomify.com/>

<sup>2</sup><http://www.starplugins.com/starzoom>



Obrázek 8.1: Ukázka nástroje Zoomify Pro.

Přiblížení lze ovládat pomocí kolečka myši, posun fotografie, pak běžným systémem drag and drop. Podobně jako Zoomify i tento nástroj počítá s předpřipravenými obrázky, o které se postará jejich nástroj **Star Zoom Atomiser**. To slouží hlavně k tomu, aby se do klientského prohlížeče posílali jen obrazová data, která jsou skutečně vidět a šetřila se tak spotřeba síťových prostředků klienta.

## Kapitola 9

# Návrh

V této části popisuji jednotlivě navržená vylepšení, která by měla vést k lepší uživatelské přívětivosti systému, ale také zefektivnit zadávání matričních záznamů.

### 9.1 Kontrola vstupů

První věcí, která značně pomůže uživateli je kontrola vstupních polí před odesláním celého formuláře na server. Nejde přitom jen o kontrolu, zda byla vyplněna povinná pole, ale zda ta vyplněná dávají smysl. Vzhledem k vyplňovaným informacím jde hlavně o pole určující čas. Tedy konkrétně zda:

- Datum narození předchází datu křtu
- Data narození osob (otec, matka, potomek) předchází datu úmrtí
- Data narození obou rodičů předchází narození potomka

Další funkcí je upozornění uživatele pokud při zadávání dat došlo k možnému překlepu. Toto lze odhalit obtížněji, a proto uživateli tuto informaci nenutíme, ale pouze jej varujeme. Upozornění je signalizováno formou ikony u popisu vstupního pole, viz obrázek. 9.1. Lze takto uživatele varovat při zadání jména či příjmení, které se doposud v databázi nenachází.



The image shows a web form with four input fields. The first field is labeled 'Jméno'. The second field is labeled 'Příjmení' and has a small warning icon (a triangle with an exclamation mark) next to the label. A dark grey tooltip box is visible over the 'Příjmení' field, containing the text 'Příjmení ještě není v Databázi jste si jisti, že nejde o překlep?'. The third field is labeled 'Vícěrčata' and has a small icon (a square with a circle inside) next to the label. The fourth field is labeled 'Pohlaví'.

Obrázek 9.1: Zobrazení varování při možné chybě.

## 9.2 Našeptávač

Další důležitou funkcí aplikace je inteligentní našeptávač. Dle částečně zadaného vstupu navrhuje uživateli zbytek slova. V polích jako je například jméno, příjmení či obec osoby, mohou být navržena předdefinovaná data nebo již zadané hodnoty z databáze 9.2 vlevo.

Pokud uživatel zadává údaje o některém z předků, systém uživateli navrhne vyplnění celé podsekce. Již při vyplnění několika údajů jako je příjmení a datum narození lze uživateli navrhnout kandidáty pro otce či matku a další předky, viz obrázek 9.2. Toto je uživateli signalizováno ikonou zvonečku. Po kliknutí se zobrazí modální okno, kde uživatel může vybrat navrženou osobu. Poté je již automaticky vyplněna celá sekce dostupnými daty. Uživatel tuto sekci může dále editovat.



Obrázek 9.2: Vlevo našeptávání křestního jména. Uprostřed modální okno pro vyplnění celé osoby. Vpravo výsledek po zvolení osoby.

## 9.3 Přehledová mapa

Samotný matriční záznam obsahuje řadu informací, které jsou ve formuláři seskupeny do logických celků. K vyplnění matričního záznamu je k dispozici vždy alespoň 20 sekcí. V takto komplexním formuláři se uživateli ne snadno orientuje a je tedy potřeba mu v tomto ulehčit.

Přehledová mapa je umístěna v levé části okna a obsahuje názvy jednotlivých sekcí formuláře. Mapa má nastavenou fixní pozici a je tak vždy vidět uprostřed okna. Aktivní sekce je zde zvýrazněna, a tak uživatel ví, ve které části formuláře se nachází. Hned tak ví, zda pro přejítí na konkrétní pole posunout okno prohlížeče nahoru či dolů.

V případě, kdy se uživatel nachází v některé z prvních sekcí a rád by z nějakého důvodu skočil na jedno z polí na konci, je mu sice informace z přehledové mapy platná, ale samotný proces je zdoluhavý. Uživatel nyní musí buď mnohokrát mačkat klávesu tab, kdy přechází mezi vstupními prvky formuláře nebo uchopit myš a skrolovat přes celý obsah okna. Proto jsou všechny položky v přehledové mapce klikací. Kliknutím na název sekce uživatele v okamžiku přeneseme na zvolené pole.



Seznam sekcí (vlevo):

- Stav
- Pozice
- Datum a adresa
- Křestitel
- Porodní bába
- Dítě
- Otec
- Otcův otec
- Otcova matka
- Otcův děda (po matce)
- Otcova bába (po matce)
- Matka
- Matčín otec**
- Matčina matka
- Matčín děda (po matce)
- Matčina bába (po matce)
- Kmotr 1
- Příbuzný kmotra 1
- Kmotr 2
- Příbuzný kmotra 2
- Břimování
- Sňatek 1
- Sňatek 2
- Datum úmrtí
- Církev

Vzorové formuláře (uprostřed):

- Matčín otec** (s ikonou otazníku): ☐ Mrtev, Jméno, Příjmení, Povolání, Obec, Ulice, Číslo popisné.
- Matčina matka** (s ikonou otazníku): ☐ Mrtvá, Jméno, Příjmení, Povolání, Obec, Ulice, Číslo popisné.
- Matčín d** (s ikonou otazníku): Jméno, Příjmení, Povolání, Obec, Ulice, Číslo popisné.

Obrázek 9.3: Přehledová mapka sekcí fomuláře.

## 9.4 Pohyb po formuláři

Jak již bylo zmíněno v předchozí podkapitole pohyb po formuláři respektive jeho jednotlivými sekcemi není ideální. Přehledová mapa odstraní některé nedostatky, ale klávesnice bude pro uživatele vždy rychlejší než myš.

Proto jsou k dispozici klávesové zkratky pro přechod mezi sekcemi, viz tabulka 9.1. Stisknutím klávesové zkratky je uživatel přenesen na první vstupní pole v dané podsekcí. Na další pole se lze dostat klasicky například pomocí tabulátoru.

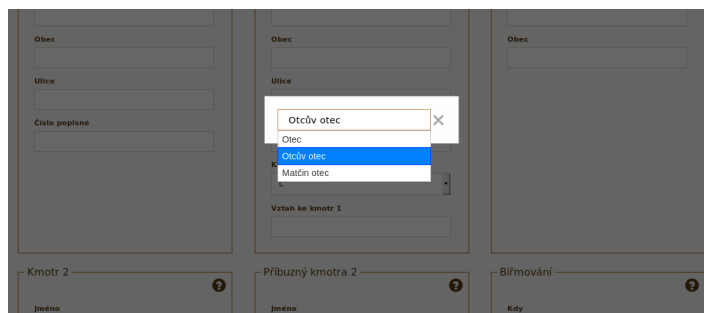
Další možností je využití klávesové zkratky CTRL + ALT + H. Ta vyvolá modální dialogové okno 9.4 s textovým vstupem, kde lze napsat název sekce, na kterou chce uživatel přejít. Přítomen je našeptávač, takže uživateli stačí zadat jen pár písmen a následně zvolit požadovanou sekci. Stejně jako v případě použití klávesových zkratk se šipkou je viditelná část stránky webového prohlížeče posunuta na zvolenou sekci.

Zkratka	Akce
CTRL + →	posun na sekci vpravo
CTRL + ←	posun na sekci vlevo
CTRL + ↓	posun na sekci pod aktuální
CTRL + ↑	posun na sekci nad aktuální

Tabulka 9.1: Klávesové zkratky pro pohyb mezi sekcemi formuláře.

## 9.5 Okno s oskenovaným záznamem

Při přepisování matričních záznamů uživatel musí vidět na originální záznam. Ten může mít otevřený v jiném okně a mezi náhledem a zadávacím formulářem přepínat. Praktičtější však je vidět obě okna současně na jednom monitoru. Dekorace okna prohlížeče a okna programu zobrazující matriční záznam zabírají zbytečně mnoho místa. Aplikace přitom neposkytuje digitalizované matriční záznamy, uživatel je však může najít na internetu [23] nebo může



Obrázek 9.4: Modální okno pro přechod na jinou sekci formuláře.

mít obrázek ve svém počítači lokálně uložen. Proto je uživateli umožněno otevřít obrázek s matričním záznamem v okně prohlížeče nad formulářem.

V pravé spodní části okna je umístěna ikona, která po kliknutí vyvolá modální okno s textovým vstupem pro zadání URL adresy obrázku. Také uživateli bude umožněno vybrat obrázek z lokálního úložiště. Po potvrzení je obrazovka rozdělena na dvě poloviny 9.5. Horní polovina obsahuje obrázek spolu s tlačítko pro zvětšení/zmenšení. Dolní polovina obsahuje formulář. V tomto zobrazení došlo ke změně rozložení podsekcí, které jsou nyní zobrazeny v jednom řádku vedle sebe. Výšku okna s obrázkem si přitom uživatel může přizpůsobit dle svých aktuálních potřeb.

Na obrázku 9.5 je naznačeno posouvání obrázku pomocí skrolování a zvětšení či zmenšení kliknutím na tlačítka. Později se však jako lepší řešení ukázalo odstranění těchto posuvníků. Ke změně přiblížení obrázku dojde při použití kolečka myši nad obrázkem. Posun obrázku za použití levého tlačítka myši, které bude simulovat dotyk. Po stisknutí levého tlačítka myši do jeho uvolnění se bude obrázek posouvat dle rozdílu pozic kurzoru od stisku a aktuální pozice.



Obrázek 9.5: Zobrazení naskenovaného matričního záznamu v okně prohlížeče.

## 9.6 Fullscreen režim

Jak je vidět například na obrázku 5.1, záhlaví aplikace zabírá relativně mnoho místa. Pokud by aplikaci chtěl používat uživatel na menším notebooku, pak by ocenil právě funkci fullscreen. Po aktivování tohoto režimu, kliknutím na příslušnou ikonu, zmizí z obrazovky jak záhlaví aplikace, tak i dekorace okna prohlížeče. Klávesou **Esc** nebo opětovným kliknutím na ikonu, uživatel tento režim opustí. Tato funkce je schopná ušetřit až 6 cm místa na displeji (dle nastavení prostředí uživatele).

## Kapitola 10

# Implementace

V této kapitole popisují jednotlivá vylepšení stávající aplikace DEMoS, která si kladou za cíl zvýšit jak efektivitu přepisování matričních záznamů, tak zpříjemnit uživatelský zážitek při samotném používání aplikace.

### 10.1 Bootstrap 4

Původní vzhled aplikace lze vidět na obrázku 5.1. Na první pohled lze vidět, že v rámci uživatelského zážitku zde byl velký prostor ke zlepšení. Rozhodl jsem se proto aplikaci upravit tak, aby co nejvíce používala komponenty z poslední verze front-end frameworku Bootstrap 4. Postupně jsem tak upravil všechny obrazovky aplikace, které byly vždy definovány ve vlastním souboru pomocí šablonovacího systému Latte.

Pro úpravu zobrazení obrazovky pro přidání a editaci matrik však bylo zapotřebí upravit chování formulářového rendereru a formulářového kontroleru. Výsledkem je pak díky Bootstrapu responzivní webová aplikace, která působí modernějším dojmem.

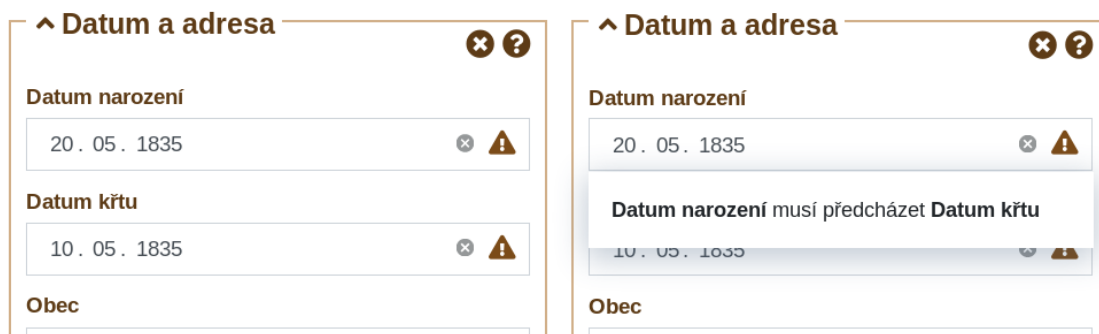
### 10.2 Kontrola vstupů

Jak již bylo uvedeno v kapitole věnující se návrhu jedno z dalších vylepšení je kontrola vstupů. Cílem je varovat uživatele pokud se mu povede zadat u osob data, které porušují chronologické pořadí. Tedy například pokud by u osoby datum úmrtí předcházelo datu narození.

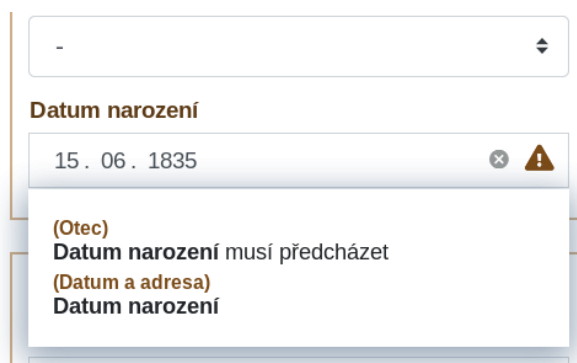
Pro kontrolu vstupů jsem se rozhodl napsat vlastní javaskriptové funkce, které lze jednoduše inicializovat tak, aby vybrané dvojice dat při změně zkontrolovaly. Pokud se po vyhodnocení vstupů zjistí chyba, je v příslušných polích zobrazena ikona výstražného trojúhelníku, který tuto nalezenou chybu uživateli signalizuje. Po najetí myši na vstup se uživateli zobrazí vysvětlující sdělení, které vstupy chybu blíže specifikuje. Toto lze vidět na obrázku 10.1, kde bylo zadáno datum křtu na dřívější datum oproti datu narození. Protože některá pole mají stejný popis, například datum narození, je třeba uživateli dát vědět, ve kterých sekcích chyba vznikla. Pokud skript vyhodnotí, že chyba je v datech z různých sekcí, jsou v chybové hlášce uvedeny názvy těchto sekcí. Lze vidět na obrázku 10.2, kde je datum narození potomka nastaveno shodně s obrázkem 10.1, navíc je však nastaveno datum narození otce až po tomto datu. Je přitom zajištěno, že pokud je způsobeno více chyb, pak se zobrazí pouze jedna chybová hláška a jedna ikona. Zároveň pokud bylo způsobeno více

chyb a je opravena pouze jedna, pak informace o dalších chybách nezmizí a uživatel tak neztrácí informaci o existenci dalších chyb.

Hlídní vstupů zajišťuje funkce `watchDateDiff` se třemi parametry. První dva definují vstupy, jejichž hodnoty chceme hlídat, třetí parametr pak definuje vlastní funkci, která je při změně volána. Tato vlastní funkce pak při změně dostává výsledek porovnání a odkaz na porovnávané vstupy. Uvnitř této funkce lze rozhodnout zda dle výsledku porovnání zavolat funkci pro přidání výstrahy nebo její odebrání. Rovněž zde lze specifikovat text chybové hlášky pro uživatele.



Obrázek 10.1: V levé části obrázku lze vidět symbolickou signalizaci chyby. Na pravé straně obrázku je vidět chybová hláška po najetí na vstupní pole.



Obrázek 10.2: Příklad zobrazení chybové hlášky, kdy jsou data z různých sekcí.

### 10.3 Přehledová mapa

Přehledová mapa byla implementována tak, jak bylo popsáno v části věnující se návrhu. Změnou však prošlo její designové provedení tak, aby ladil se zbytkem aplikace. Nachází se na pravé straně okna a její maximální výška je 80 procent velikosti uživatelského displeje, pokud tato velikost nestačí, je přítomen posuvný prvek pro zobrazení zbytku položek. V mapě je automaticky zvýrazněna aktivní sekce formuláře a je-li potřeba je upraven posun v rámci této mapy tak, aby byl tento prvek vždy vidět v polovině výšky mapy. Kliknutím na libovolnou položku této mapy je uživatelské posunuto na zvolenou sekci a ta se stává aktivní. Uživatel tak při rychlém pohledu na mapu ví, které sekce formuláře se nachází okolo právě

vyplňované sekce. Nastavení pozice posuvného prvku přehledové mapy na pravou stranu, kde končí okno je umožněno díky CSS vlastnosti pro specifikaci směru textu na hodnotu `rtl` (right-to-left). Výsledný vzhled je k vidění na obrázku 10.3. Pokud by uživatel disponoval pouze menším displejem a tedy menší šířkou okna prohlížeče, je přehledová mapa skryta.

Obrázek 10.3: Přehledová mapa formuláře.

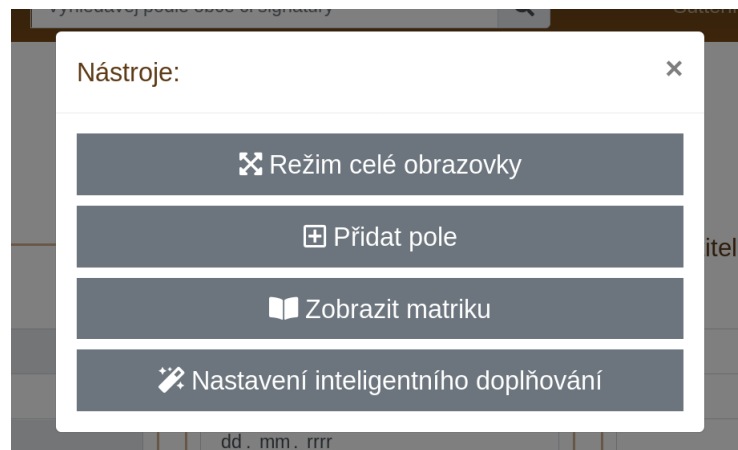
## 10.4 Sjedené umístění nástrojů

Původní verze aplikací disponovala jedinou funkcí a to přidání polí pro další kmotry a jejich příbuzné. Po kliknutí na ikonu v pravé spodní části obrazovky, viz obrázek 5.1, se zobrazilo okno kde si uživatel zvolil kolik takovýchto polí chce do formuláře přidat. V této aplikaci je nutné počítat s rostoucím počtem dalších funkcí a proto jsem nahradil původní symbol pro přidání, ikonou symbolizující nástroje. Toto tlačítko vyvolá modální okno kam lze přidat další uživatelské funkce. Vyvolané modální okno je vidět na obrázku 10.4, kde jsou vidět čtyři funkce. Kliknutí na tlačítko aktivuje danou funkci nebo vyvolá další modální okno pokud je potřeba od uživatele další vstup. Povedlo se tak sjednotit umístění všech nástrojů, které mohou být i v budoucnu vytvořeny a uživatel tak bude vědět, na kterém místě je hledat.

## 10.5 Pohyb po formuláři

Pro snazší a rychlejší pohyb mezi sekcemi formuláře byly vytvořeny kromě přehledové mapy ze sekce 10.3, také další dva možné způsoby navigace bez použití myši navržených v sekci 9.4. Uživatel je o těchto způsobech pohybu informován pomocí nápovědy na počátku formuláře.

Klávesové zkratky pro pohyb mezi sekcemi jsou implementovány pomocí JavaScriptu a zachycení událostí stisku klávesnice. Protože aplikace je díky použití Bootstrapu 10.1 responzivní a uživatel má možnost posunu mezi sekcemi nejen doprava či doleva, ale také nahoru i dolů. Dle velikosti okna prohlížeče tak může být na jednom řádku jedna, dvě nebo tři sekce. Skript toto zohledňuje, aby posun mezi sekcemi odpovídal aktuálnímu rozložení.



Obrázek 10.4: Modální okno s užitečnými funkcemi.

Druhou možností je vyhledání sekce vyvoláním modálního okna klávesovou zkratkou. Toto okno obsahuje jediný textový vstup spolu s našeptávačem sekcí formuláře. Po zvolení sekce je uživatel okamžitě na tuto sekci přenesen.

## 10.6 Zobrazení digitalizované matriky

V kapitole 4 jsem popisoval stav digitalizovaných matričních záznamů v České republice. Žádný ze zmíněných archivů v současnosti bohužel neposkytuje API, které by dovolovalo získat konkrétní stranu konkrétní matriky. Pokud se ale situace v budoucnu změní, lze pak snadno přidat funkcionalitu, která by dovolovala uživateli zobrazit matriční sken bez nutnosti zadávat URL adresu, neboť potřebné metody implementovaná třída obsahuje.

Do té doby uživatel může využít funkce zobrazení digitalizované matriky přímo v okně s formulářem zadáním zdrojové adresy obrázku. Po kliknutí na nástrojovou ikonu ze sekce 10.4 se zobrazí modální okno s výběrem funkcí, viz obrázek 10.4. Následně uživatel pouze zvolí funkci "Zobrazit matriku" a zadá patřičnou URL adresu. Uživatel má navíc před potvrzením URL adresy možnost volby, zda chce využít i navigační miniaturu obrázku. Jedná se o stejnou funkci známou z pokročilejších javaskriptových knihoven popsanych v kapitole 8 u prémiových verzí knihovny Zoomify. Uživateli se tak nad formuláři zobrazí zvolený obrázek s možností jej zvětšit či zmenšit kolečkem myši a pomocí gesta drag and drop obrázkem posunout do stran. Pokud uživatel nezvolil jinak, je v levém horním rohu přítomná miniatura s obdélníkem s poměrem stran korespondujícím s velikostí hlavního obrázku. Výšku plochy, kterou obrázek v okně zabírá může uživatel měnit a zobrazení si tak více přizpůsobit. Obdélník na miniatuře vyznačuje, která část obrázku je zobrazená, mění se tedy jeho velikost v závislosti na přiblížení hlavního obrázku. Tím lze posunout i gestem drag and drop nad tímto obdélníkem. Příklad výsledného zobrazení je na obrázku 10.5

Samozřejmostí je i tlačítko v pravém horním rohu které obrázek s matrikou opět skryje. Tlačítko pro skrytí obrázku i miniatura jsou uživateli zobrazeny pouze po najetí myši, aby zbytečně nepřekáželi. Implementována je o ochrana, která zamezí uživateli posunout obrázek mimo vymezený rám pro tento obrázek. Nemůže se tak stát, že by uživatel obrázek posunul mimo rám a již jej nemohl posunout zpět.





Obrázek 10.6: Rozložení při režim celé obrazovky spolu se snímkem matriky.

## 10.8 Vlastní rozvržení formuláře

Jak bylo popsáno v kapitole 5, tato aplikace je součástí aktivně vyvíjeného projektu. Vznikají tak v průběhu nové požadavky a jedním z nich bylo i vylepšení a rozšíření funkce pro přidání sekcí formuláře. Cílem tohoto rozšíření je umožnit uživateli více si přizpůsobit rozvržení formuláře změnou pořadí jednotlivých sekcí, a také možnost volby vybrané sekce úplně skryt.

Pro změnu pořadí sekcí bylo využito HTML5 událostí `dragstart` `dragover` `dragleave` `drop`. Díky těmto událostem lze implementovat změnu rozložení uživateli známým gestem drag and drop. Na obrázku 10.7 lze vidět jak přesun sekce probíhá. Uživatel uchopí řádek reflektující danou sekci a může jej upustit na libovolné umístění v seznamu sekcí. O výsledném umístění sekce je uživatel informován přerušovanou čarou, která se objevuje v závislosti na poloze ukazatele, mezi jednotlivými řádky. Na pravé straně každého řádku lze vidět zaškrtnávací tlačítko, to slouží ke skrytí nebo naopak zobrazení sekce formuláře.

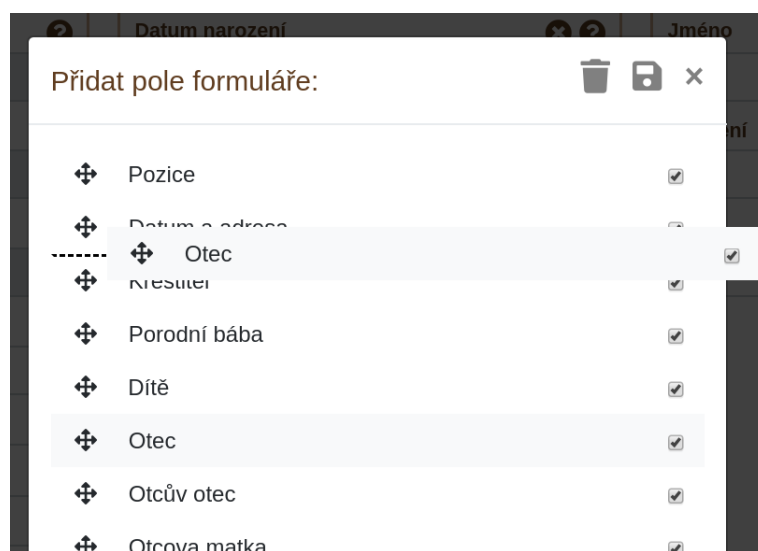
V záhlaví modálního okna lze vidět kromě tlačítka pro uzavření tohoto okna také další dvě tlačítka. Tato tlačítka slouží k uložení aktuálního rozložení a obnovu původního rozložení. Zde je využito úložiště zvané `localStorage`. Hlavní rozdíl oproti jiným typům úložišť, které je v rámci prohlížeče možné využít, jako je například `sessionStorage` či známé `cookies`, nemá toto úložiště pro data expirační čas. To znamená, že po uložení dat je i po zavření okna prohlížeče a pozdějšímu znovu navštívení aplikace možné k datům přistupovat. Díky tomu si uživatel může jednou nastavit vlastní rozvržení sekcí formuláře a při přepisování dalšího záznamu je mu toto rozvržení zachováno.

Po kliknutí na ikonu uložit skript projde všechny sekce formuláře v pořadí odpovídajícím DOM struktuře a poznačí si jejich viditelnost. Tato data jsou v `localStorage` uloženy jako

dvojice název a hodnota. Hodnotou je v tomto případě výše popsaná data popisující aktuální rozvržení formuláře ve formátu JSON. Samotný JSON má jednoduchou strukturu dvojic textový klíč a hodnota. Klíčem je HTML identifikátor dané formulářové sekce, hodnota je pak boolovského typu s hodnotou odrážející viditelnost této sekce.

Po načtení stránky skript zkontroluje existenci těchto dat a případně pomocí metody `Node.insertBefore()` změní uspořádání sekcí a také těmto sekcím nastaví viditelnost. Původní rozložení tak již nelze z přeskádané struktury DOM vyčíst. Jendou z možností jak se poté vrátit k původnímu rozložení je smazat data o rozložení z `localStorage` a obnovit stránku. To by ale zapříčinilo ztrátu dat ve formuláři pokud by uživatel již některé položky vyplnil. Proto si skript před úpravou struktury formuláře, vyvolaný metodou `load()` třídy obstarávající manipulaci s rozvržením sekcí formuláře, poznačí stejný formát dat, který ukládá do `localStorage`. Protože se tak děje před přeskládáním hovoří tak tyto data o původním rozvržení sekcí. Při kliknutí na ikonu koše, se tak tato data s původním rozvržením sekcí zapíše do `localStorage` a poté se zavola metoda `load()`, která toto původní rozložení aplikuje.

Poslední věcí, kterou je potřeba zabezpečit, je aby změny pořadí a viditelnosti provedené uživatelem byly reflektovány v přehledové mapě ze sekce 10.3. To je zajištěno voláním funkce, která obsah přehledové mapy vždy obnoví.



Obrázek 10.7: Modální okno pro změnu rozvržení formuláře.

## 10.9 Automatické doplnění záznamu

Hlavní podpůrnou technikou pro zefektivnění přepisu matrik je funkce automatické doplnění. V matričních knihách je běžné, že se na několika záznamech vyskytují stejné osoby. Například se často opakují porodní báby, křtitelé či kmotři. U sourozenců se navíc opakují rodiče a další rodinní příslušníci. Pokud tedy již existuje záznam o osobě, která by se mohla objevit v uživatelem aktuálně přepisovaném záznamu, je vhodné mu práci ulehčit a takovouto osobu za něj vyplnit. Protože cílem této aplikace je přepis matrik a dalších serilních pramenů, tak jak byly prve napsány je třeba, aby uživatel o každém doplnění ze strany

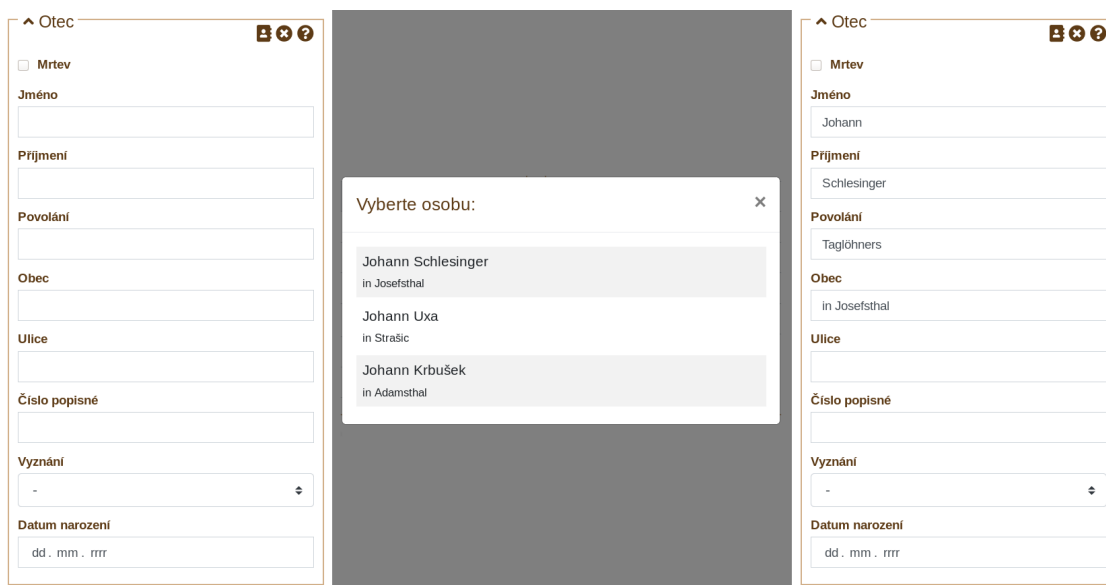
aplikace věděl. Je tedy nutné, aby byl buď o každém doplnění informován nebo se tak dělo na základě jím potvrzené akce. Pouze takto docílit toho, že uživatel ví, že má doplněná data zkontrolovat.

Automatické doplnění je v této kapitole rozděleno do dvou částí, které vycházejí z návrhu v sekci 9.2. V první části je popsána funkce doplnění celých osob. Ve druhé pak inteligentní našeptávač jednotlivých údajů formuláře.

### 10.9.1 Návrh osob

Cílem této funkcionality je navrhnout uživateli jednoduché vyplnění celé sekce formuláře. Při načtení formuláře a každé další změně některého vstupního pole se na server zasílá JSON zpráva. Ta obsahuje informaci o všech již vyplněných polích a jejich hodnotách. Na serveru se poté dle těchto dat hledá v databázi relevantní osoby. Dle osoby, pro kterou se návrh vyhledává se zohledňují různě informace.

Po té co server získá seznam osob pro návrh, jsou data zaslána zpět do prohlížeče klienta opět ve formátu JSON. JavaScript tuto odpověď zachytí a zpracuje. Ve formulářových sekcích, pro které existuje návrh osob vytvoří v záhlaví ikonu. Po kliknutí na tuto ikonu se uživateli zobrazí modální okno pro výběr osob. Na každém řádku jsou vypsány dostupné informace o dané osobě, na základě kterých se uživatel rozhodne který návrh použít. Kliknutím na řádek návrhu se modální okno uzavře a formulářová sekce se vyplní dostupnými údaji.



Obrázek 10.8: Vlevo v záhlaví lze vidět ikonu signalizující možnost doplnění osoby(první ikona zleva). Uprostřed modální okno s výběrem osob. Vpravo výsledek doplnění osoby.

### 10.9.2 Našeptávač

Poslední funkcí, která má snahu uživateli urychlit vyplnění záznamu je našeptávač. Ten se zobrazí, jakmile uživatel začne do pole psát. Obsahuje možné hodnoty pro daný vstup, které se již v databázi nachází dle největší četnosti.

Inspiroval jsem se přitom našeptávači popsanými v sekci 7.2 a 7.3. Pro základní funkcionality našeptávače byly použity možnosti komponenty jQuery UI ze sekce 6.5.1. Ty jsem rozšířil přepsáním metod pro výpis položek našeptávače, což umožnilo mezi první položky našeptávače přidat návrh celých osob. Pokud tak dle dostupných dat je k dispozici návrh osoby, pak jej uživatel může velmi rychle zvolit z našeptávače a vyplnit tak i celou sekci. Na obrázku 10.9 lze vidět, že uživatel napsal první písmeno ze jména otce. Našeptávač mu již nabízí, krom jmen začínajících na stejné písmeno i tři podrobnější záznamy osob s dalšími daty. První položka našeptávače obsahuje kromě jména také příjmení, povolání a obce osoby. Druhá položka nenabízí příjmení, avšak obsahuje datum narození, obci a číslo popisné.

The image shows two side-by-side screenshots of a web form titled "Otec".

**Left Screenshot:** The form has a title "Otec" and a checkbox "Mrtev". The "Jméno" (Name) field contains the letter "J". A dropdown menu is open, showing a list of suggestions. The first suggestion is "Johann Uxa - Platzmeister in Josefsthal". The second suggestion, "Josef - 1860-03-15 Adamsthal 13", is highlighted in brown. Other suggestions include "Josefa Blažek von Babitz", "Josefa", and "Josef". Below the dropdown are fields for "Číslo popisné" (Cadastral number), "Vyznání" (Religion) with a dropdown showing "-", and "Datum narození" (Date of birth) with a placeholder "dd . mm . rrrr".

**Right Screenshot:** This shows the form after the second suggestion has been selected. The "Jméno" field now contains "Josef". The "Příjmení" (Surname) field is empty. The "Povolání" (Occupation) field is empty. The "Obec" (Municipality) field contains "Adamsthal". The "Ulice" (Street) field is empty. The "Číslo popisné" field contains "13". The "Vyznání" field now shows "bez vyznání" (no religion). The "Datum narození" field now contains "15 . 03 . 1860" and has a small 'x' icon to clear the field.

Obrázek 10.9: V levé části našeptávač pro pole křestního jména. Vpravo výsledek doplnění po vybrání druhé položky.

# Kapitola 11

## Testování

Kapitola se věnuje testování výsledného efektu implementovaných podpůrných technik na úsporu času pro přepis matrik. Nejprve popisují způsob testování a popis použitého scénáře. Následují výstupní data získaná z testování a nakonec se zaměřují na úpravu jedné z funkcí na základě podnětu získaného z testování.

### 11.1 Uživatelské testování

Aktuální verze projektu DEMoS byla v době testování přístupná na serveru VUT FIT<sup>1</sup>, kde ji testovala skupina genealogů. Byl vypracován scénář, který zahrnoval přepis matriky nejprve bez mnou implementovaných podpůrných funkcí. Následně se genealogové seznámili s funkcemi a vytvořili další záznam s možností použít veškeré funkce. Nakonec genealogové vyplnili dotazník, kde zhodnotili čas, který jim zabrali obě varianty při přepisu záznamu, a také hodnocení přínosu jednotlivých funkcí.

Na závěr mohli genealogové v dotazníku uvést vlastní poznatky k aplikaci a také uvést jakou další funkci by v aplikaci ocenili. Podrobnější výstup tohoto testování shrnuji v následující sekci.

### 11.2 Výsledky testování

Skupina genealogů provádějící testování se skládala z pěti členů. Porovnání časů potřebných k vyplnění matričního záznamu lze vidět na grafu z obrázku 11.1. Z grafu lze vypožorovat, že u všech uživatelů došlo při použití podpůrných technik ke zrychlení přepisu záznamu. V průměru se přepis záznamu zkrátil o 2,2 minuty.

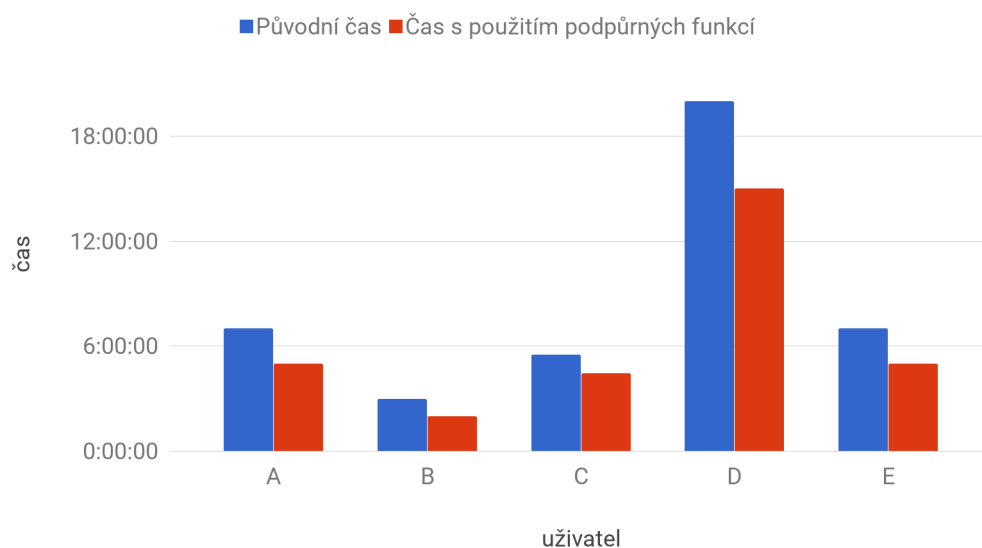
K hodnocení užitečnosti jednotlivých funkcí mohli uživatelé vybrat číslo v rozmezí od jedné do pěti, kdy pět znamenalo nejvíce užitečné. Graf na obrázku 11.2 zobrazuje toto průměrné hodnocení.

Uživatelé také měli prostor pro vlastní poznámku k aplikaci, ze kterých vyplynulo toto:

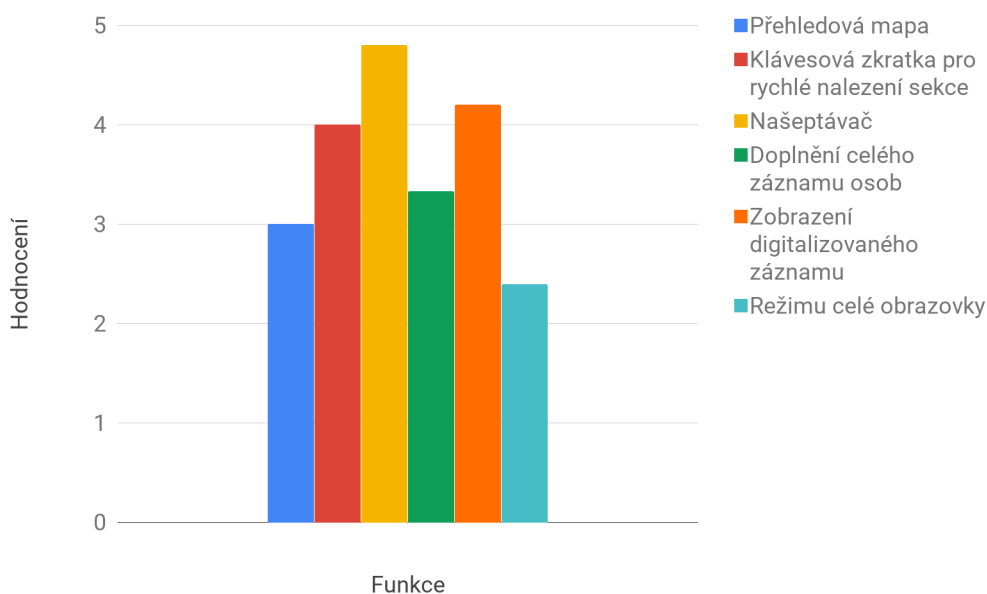
- Aplikace je vyhovující, ale uvítala bych líbivější vzhled.
- Přehledová mapa by práci určitě o dost urychlila, na standardním notebooku se ale nezobrazuje.
- Hodil by se digitalizovaný záznam, který odpovídá matrice.

---

<sup>1</sup><http://perun.fit.vutbr.cz/demos/>



Obrázek 11.1: Porovnání časů k vyplnění matričního záznamu dle výsledků dotazníku.

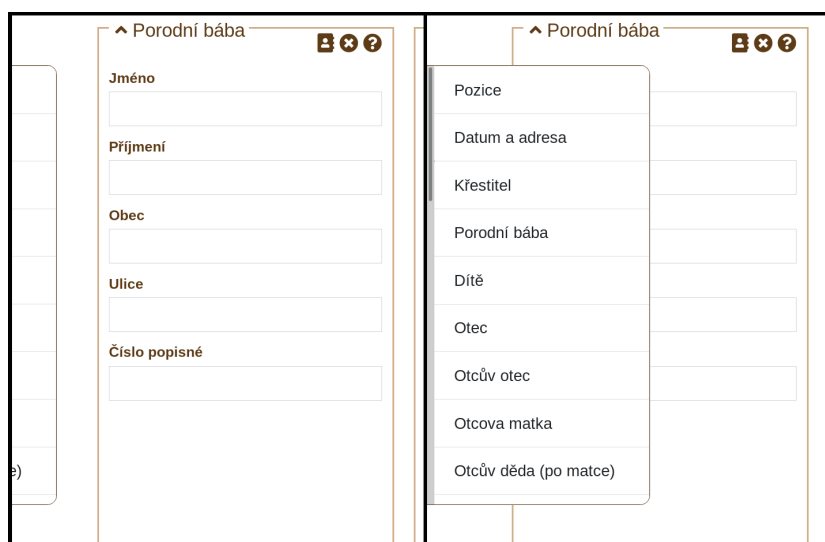


Obrázek 11.2: Hodnocení jednotlivých funkcí uživateli.

### 11.3 Zhodnocení

První poznámka z předchozí sekce hodnotí vzhled samotné aplikace, kde je jistě stále prostor pro další vylepšení. Zaměřil jsem se však hlavně na druhou připomínku. Jak bylo popsáno v sekci 10.3 v rámci responzivního vzhledu aplikace, se přehledová mapa skryje pokud pro ní na displeji již není dostatek místa.

Tento požadavek jsem řešil tak, že přehledová mapa je při užším okně z 80% skryta nastavením CSS vlastnosti `transform: translateX(-80%)`. Celou přehledovou mapu uživatel zobrazí najetím myši na její viditelnou část. Výsledek této úpravy lze vidět na obrázku 11.3. Uživatel tak funkce přehledové mapy může využít i na zařízeních s menšími displeji, aniž by mu mapa překážela.



Obrázek 11.3: Vlevo částečně skrytá přehledová mapa. Vpravo plně zobrazena po najetí myši.

Poslední poznámka uživatele poukazuje na skutečnost, že pro zobrazení obrázku digitalizované matriky musí být zadána URL adresa obrázku. Aplikace DEMoS neobsahuje digitalizované matriky a ani s touto funkcí nelze do budoucna počítat. Bohužel však v současnosti neexistuje ani jediný archiv disponujícím API, které by umožnilo dle identifikátoru matriky její obrázek získat. Je však možné, že takovéto API vznikne a v budoucnu pak na jeho základě bude možné uživateli zobrazit správný obrázek matriky stisknutím jediného tlačítka.

## Kapitola 12

# Závěr

Cílem této práce bylo seznámení se s typy a strukturou matričních záznamů a podpůrných technik pro jejich přepisování. Na základě nastudovaných technik poté navrhnout vylepšení stávající webové aplikace DEMoS pro přepis matričních záznamů, která povedou k co efektivnějšímu přepisu matrik do počítače se zaměřením na automatické doplnění slov, rozložení GUI a pohyb v něm.

K vytvoření výsledného řešení bylo nutné nastudovat řadu různých webových technologií a seznámit se s jinými řešeními daných problému. Bylo poté implementováno několik vylepšení, která zkvalitňují uživatelský zážitek z aplikace a zefektivňují přepis matričních záznamů.

Výsledky této práce dále byly testovány na skupině genealogů pro zhodnocení efektivity pro přepis jednoho záznamu a užitečnost implementovaných funkcí. Ty potvrdily nezanedbatelnou časovou úsporu a zkvalitnění uživatelského zážitku. Testování také odhalilo slabinu jedné z implementovaných funkcí, která poté byla upravena.

Podpůrné techniky, které byly vytvořeny pro přepis matrik se dají s minimálními úpravami použít při rozšíření aplikace DEMoS o přepis dalších serilních pramenů.



# Literatura

- [1] Blanka, L.: *Sestavte si rodokmen: pátráme po svých předcích*. Grada, 2012, ISBN 978-80-247-4069-0.
- [2] Consortium, W. W. W.: Autofill. [Online; navštíveno 4.01.2019].  
URL <https://www.w3.org/TR/html52/sec-forms.html#autofilling-form-controls-the-autocomplete-attribute>
- [3] Crockford, D.: Introducing JSON. [Online; navštíveno 6.02.2019].  
URL <https://www.json.org/>
- [4] Czernín, D.: *Uživatelské rozhraní pro přepis matrik*. Říjen 2018, [Online; navštíveno 27.10.2018].  
URL [https://www.vutbr.cz/www\\_base/zav\\_prace\\_soubor\\_verejne.php?file\\_id=181146](https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=181146)
- [5] Digitální badatelna Archivu města Ostravy: Digitální badatelna Archivu města Ostravy. [Online; navštíveno 23.11.2018].  
URL <https://badatelna.ostrava.cz/main>
- [6] FamilySearch: About FamilySearch. [Online; navštíveno 7.10.2018].  
URL <https://www.familysearch.org/home/about>
- [7] Fornés, A.; Lladós, J.; Mas, J.; aj.: A bimodal crowdsourcing platform for demographic historical manuscripts. In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, ACM, 2014, s. 103–108.
- [8] Google: Autocomplete for Addresses and Search Terms. [Online; navštíveno 10.01.2019].  
URL <https://developers.google.com/maps/documentation/javascript/places-autocomplete>
- [9] Hankovec, M.: DigiArchiv SOA v Třeboni. [Online; navštíveno 23.11.2018].  
URL <https://digi.ceskearchivy.cz/uvod>
- [10] Hanousek, T.: Archiv hlavního města Prahy. [Online; navštíveno 23.11.2018].  
URL <http://www.ahmp.cz/>
- [11] JS Foundation: jQuery. [Online; navštíveno 26.10.2018].  
URL <https://jquery.com/>
- [12] Mark Otto, J. T.; contributors, B.: Bootstrap. [Online; navštíveno 26.10.2018].  
URL <https://getbootstrap.com>

- [13] Moravský zemský archiv v Brně: Lánové rejstříky. [Online; navštíveno 10.11.2018].  
URL <http://www.mza.cz/a8web/a8apps1/D1/A8SL4DD2Bad3D1.htm>
- [14] Moravský zemský archiv v Brně: Moravský zemský archiv v Brně. [Online; navštíveno 23.11.2018].  
URL <http://www.mza.cz/>
- [15] Moravský zemský archiv v Brně: Opisy urbářů. [Online; navštíveno 11.11.2018].  
URL <http://www.mza.cz/a8web/a8apps1/G371/A8SL4DD2Bad3G371.htm>
- [16] Moravský zemský archiv v Brně: Rektifikační akta. [Online; navštíveno 10.11.2018].  
URL <http://www.mza.cz/a8web/a8apps1/D2/A8SL4DD2Bad3D2.htm>
- [17] Moravský zemský archiv v Brně: Sčítací operáty. [Online; navštíveno 11.11.2018].  
URL <http://www.mza.cz/a8web/a8apps1/so/A8SL4DD2Bad3S0.htm>
- [18] Moravský zemský archiv v Brně: Urbáře. [Online; navštíveno 11.11.2018].  
URL <http://www.mza.cz/a8web/a8apps1/URB01/A8SL4DD2Bad3URB01.htm>
- [19] Nette Foundation: Nette Framework. [Online; navštíveno 13.10.2018].  
URL <https://nette.org/>
- [20] Národní archiv: Národní archiv. [Online; navštíveno 23.11.2018].  
URL <http://www.nacr.cz/>
- [21] Pecháček, J.: Jak vypadají matriční záznamy. [Online; navštíveno 4.10.2018].  
URL <https://www.odkudjsme.cz/blog/jak-vypadaji-matricni-zaznamy/>
- [22] Podhola, J.: Matriky. [Online; navštíveno 27.10.2018].  
URL <http://www.genea.cz/informace/badani-v-archivu/matriky/>
- [23] Česká genealogická a heraldická společnost v Praze: Digitalizace. [Online; navštíveno 28.10.2018].  
URL <http://www.genealogie.cz/aktivity/digitalizace/>
- [24] Státní oblastní archiv Litoměřice: Státní oblastní archiv v Litoměřicích. [Online; navštíveno 23.11.2018].  
URL <http://www.soalitomerice.cz/>
- [25] Státní oblastní archiv v Plzni: Státní oblastní archiv v Plzni. [Online; navštíveno 23.11.2018].  
URL <http://www.soaplzen.cz/>
- [26] Vysoké učení technické v Brně: Možnosti tvorby komunitní genealogické databáze se sémantickou informací a zahrnutou neurčitostí. [Online; navštíveno 3.11.2018].  
URL <https://www.vutbr.cz/vav/projekty/detail/29197>
- [27] Zemský archiv v Opavě: Zemský archiv v Opavě. [Online; navštíveno 23.11.2018].  
URL <http://www.archives.cz/web/>

## Příloha A

# Obsah přiloženého paměťového média

- demos/ - zdrojové kódy aplikace DEMoS
- db/ - vyexportovaná databáze
- README - návod ke spuštění systému
- tex/ - zdrojové kódy textové části této práce pro L<sup>A</sup>T<sub>E</sub>X